AL/HR-TP-1993-0026

AD-A272 186

SUCCESSIVELY APPROXIMATING HUMAN PERFORMANCE

Michael J. Young

HUMAN RESOURCES DIRECTORATE
LOGISTICS RESEARCH DIVISION
2698 G STREET
WRIGHT-PATTERSON AFB, OH 45433-7604

SEPTEMBER 1993

INTERIM TECHNICAL PAPER FOR PERIOD OCTOBER 1992 TO MAY 1993

Approved for public release; distribution is unlimited

A R M S T R O N G   L A B O R A T O R Y

AIR FORCE MATERIEL COMMAND
WRIGHT-PATTERSON AIR FORCE BASE, OHIO 45433-6573

NOTICES

This paper has been reviewed and is approved for publication.

Michael J. Young

MICHAEL J. YOUNG
Contract Monitor

BERTRAM W. CREAM, Chief
Logistics Research Division

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY (*Leave blank*)	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	September 1993	Interim - Oct 1992 to May 1993

4. TITLE AND SUBTITLE
Successively Approximating Human Performance

5. FUNDING NUMBERS
C - F33615-91-D-0009
PE - 62205F
PR - 1710
TA - 00
WU - 60

6. AUTHOR(S)
Michael J. Young

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Armstrong Laboratory
Human Resources Directorate
Logistics Research Division
2698 G Street
Wright-Patterson AFB, OH 45433-7604

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
Armstrong Laboratory
Human Resources Directorate
Logistics Research Division
2698 G Street
Wright-Patterson AFB, OH 45433-7604

10. SPONSORING/MONITORING AGENCY REPORT NUMBER
AL/HR-TP-1993-0026

11. SUPPLEMENTARY NOTES
Armstrong Laboratory Technical Monitor: Michael J. Young, AL/HRGA, (513) 255-8229

12a. DISTRIBUTION / AVAILABILITY STATEMENT
Approved for public release; distribution is unlimited

12b. DISTRIBUTION CODE

13. ABSTRACT (*Maximum 200 words*)
This report discusses research issues identified by two National Research Council studies on human performance models and proposes a new approach to model development, called the method of successive approximation, to address these issues. Successive approximation is an incremental approach to human performance process model development. During each increment of development, a complete human performance model is built. Initial models, however, are limited in the behavior they can represent. Subsequent iterations of development extend the behavioral repertoire by increasing the resolution of the human performance model. In addition, the report discusses issues of psychological reality as they relate to human performance models.

14. SUBJECT TERMS
cognitive architecture
cognitive psychology
human performance model

15. NUMBER OF PAGES
39

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	SAR

NSN 7540-01-280-5500

Standard Form 298 (Rev 2-89)
Prescribed by ANSI Std Z39-18
298-102

CONTENTS

FIGURES

Preface

This report discusses research issues identified by two National Research Council studies on human performance models (HPMs) and proposes a new approach to model development--the method of successive approximation--to address these issues. Successive approximation is an incremental approach to human performance process model development. During each increment of development, a complete HPM is built. Initial models, however, are limited in the behavior they can represent. Subsequent iterations of development extend the behavioral repertoire by increasing the resolution of the HPM. In addition, this report discusses issues of psychological reality as they relate to HPMs.

This work supports the Armstrong Laboratory, Logistics Research Division Acquisition Logistics Branch's (AL/HRGA) advanced development Operability Assessment System program (work unit number 2940-03-02) and was accomplished by the author under the Research, Development, Training and Evaluation Support work unit (1710-00-60). The author extends his to thanks to Tom Killion and Kate McCreight for reading earlier drafts of the paper and making useful suggestions for improving the substance and readability of the text. The author also thanks his colleagues at BBN Systems and Technologies--Dick Pew, Shelly Baron, Carl Feehrer, Steve Deutsch, Marilyn Adams, and Eva Hudlicka--for many stimulating discussions on human performance modeling. The ideas herein owe much to those conversations.

SUCCESSIVELY APPROXIMATING HUMAN PERFORMANCE

INTRODUCTION

In recent years, the National Research Council (NRC) has sponsored two studies on human performance models (HPMs). The first study, commissioned by the National Aeronautics and Space Agency (NASA) Ames Research Center, reviewed HPMs to determine if a set of models existed that would support development of an HPM for a human-centered computer-aided engineering (CAE) facility. Specific study objectives included reviewing current HPMs, identifying those models that would be most useful for a CAE facility, identifying limitations of these models, providing guidance on model usage, and recommending research to overcome model limitations. This study group's report, titled "Human Performance Models for Computer-Aided Engineering" (Elkind et al., 1989), lists several findings related to HPMs.

The first finding addresses types of models and desired attributes. The study group strongly prefers computational models (i.e., those that can be implemented in a computer) over noncomputational models. The rationale is that computational models are better at identifying the stimulus and response linkages needed to illustrate the effects of system design on human performance. In addition, the study group advocates that models be explicit in their required inputs and outputs to facilitate use.

A second finding addresses the adequacy of existing models. The study group reviewed several hundred HPMs and found that existing models do not provide a complete description of human perceptual and cognitive performance. From a perspective of stages of information processing, there were significant gaps in the availability of models for several stages. Furthermore, linkages between models representing different stages within a modality (e.g., vision) are weak and linkages between modalities (e.g., vision and cognition) are essentially nonexistent. Finally, the study group found no available framework to integrate existing models.

A third finding addresses the lack of techniques or methods for validating HPMs. The report emphasizes that model validation must be an integral part of any effort to develop HPMs and model validation must be a continuous effort.

A final finding of the study group is that building HPMs and employing them in the system design process require the creation of a modeling framework consisting, in part, of a toolbox to help model developers and designers create models, run them in a simulation, and analyze their output. In addition, the framework should provide guidance to help developers and designers determine when to use HPMs, what data is needed to run these models, and when results are valid.

The second HPM study was commissioned to develop recommendations for improving the utility of HPMs. Specific study objectives included evaluating the strengths and weaknesses of alternative modeling approaches, assessing the conditions under which current models are of practical use, assessing the potential for developing comprehensive HPMs, and recommending research or other courses of action to improve human performance modeling. The second study group's report, titled "Quantitative Modeling of Human Performance in Complex, Dynamic Systems" (Baron et al., 1990), identifies eight research issues associated with extending the scope and applicability of HPMs.

The first research issue is the development of complex/comprehensive models. Traditionally, HPMs are designed for situations in which an operator is engaged in a single task controlling a machine. In most real-world situations, however, the system operator is engaged in complex tasks requiring multiple inputs and outputs. In addition, operators usually have multiple

goals which are frequently interrupted. Modeling of real-world tasks demands the development of more sophisticated models. The report lists two recommendations: (1) develop computational models of attention, because attention will be a key component of any comprehensive model; and (2) explore the development of models that integrate single-task models into a comprehensive model.

The second issue is model parameterization. The report notes that as models become more complex (i.e., incorporate more parameters) the data necessary to specify the parameters becomes more difficult and costly to collect. Two of the report's recommendations are: (1) identify and classify all model parameters, and (2) develop estimation techniques to uncover the distribution of parameter values.

The third research issue is model validation. The report points out that as models become more complex they become more difficult and costly to validate. The lack of validated models is the main limiting factor to a wider use of HPMs in system design. The report recommends the development of new methodologies to validate models.

The fourth research issue is the under utilization and inaccessibility of most HPMs. According to the report, most complex HPMs have not been widely used or subjected to independent evaluation. The report recommends that government agencies support the development of easily used versions of models on the most inexpensive machines possible. In addition, the report advocates the development of user-friendly interfaces and good documentation to facilitate model use.

The fifth research issue is the potential for misuse and misunderstanding of models as they become more complex. One of the report's recommendations is for better documentation of model assumptions and theoretical bases through published papers. The report also advocates the development of good documentation describing the embedded data and software requirements necessary for model usage.

The sixth research issue is the lack of models accounting for the cognitive components of tasks. The report notes the increasing role of cognitive activities in managing complex systems and the general lack of models accounting for cognitive behavior. (Most contemporary models have focused on modeling perceptual or psychomotor components of behavior.) The report recommends the development of new techniques and methods to account for the operator's mental representation.

The seventh issue concerns the immaturity of knowledge-based modeling techniques. The report notes that knowledge-based models appear well suited for implementing cognitive models. However, at the time of the report, very little work had actually been completed towards the development and testing of knowledge-based models. The report recommends that additional effort be focused on creating methodological approaches to support the development and application of knowledge-based models, and on validating this approach to modeling cognitive functions.

The final modeling issue discussed in the report is accounting for individual differences in HPMs. To date, individual differences have largely been ignored in favor of normative or average indices. The report notes that due to the increased complexity of new systems, individual differences will play an increasingly important role in man/system performance. The report recommends that HPMs be used to assess the role of individual differences in overall system performance.

A process HPM is being developed to address these issues identified by the NRC studies. Human performance process (HPP) models are engineering-like models (Corker, 1991; Young,

1992) which emulate human behavior by simulating specific human information processing attributes and processes. HPP models represent the human information processing system as an assemblage of input and output devices, processors, and memory-storage subsystems. Information flow and transformation through the model is described in terms of key parameters which denote processing limitations of the subsystems.

There are two intellectual antecedents to HPP model development: engineering psychology and cognitive psychology. Engineering psychology investigates the implications that human information-processing capabilities have for system design (Wickens, 1992). Engineering psychologists are the most prolific developers of HPMs. The goal of engineering psychologists is to develop HPMs with considerable predictive power. Engineering psychologists, through the use of experimental studies, strive to define mathematical relationships between input and output variables. Functional relationships between input and output variables described in mathematical terms are called analytic models.

Analytic models, however, are difficult to generalize. A scientist wishing to employ an analytic model in a design study must determine if the model is valid for the task to be studied. To make this determination, some dimensions of the new task are compared to the task on which the analytic model was originally validated. If the two tasks are similar enough, the model can be used; if the tasks are dissimilar, either additional experimental studies can be performed to validate the model or a different model must be found.

There is no general agreement about which task dimensions are important in comparing tasks. Some authors propose a criterion-measures approach, others an information-theoretic approach, others a task-strategies approach, and still others an ability-requirements approach (cf. Fleishman & Quaintance [1984] for a discussion of the issues). The end result of this lack of agreement on a standard taxonomy for tasks is that models developed for one task (or set of tasks) are rarely used for other tasks; rather, scientists develop new models for each new application. This has led to a proliferation of analytic models, each relevant only in a very limited domain.

Analytic models do not describe the mechanisms underlying performance; they only capture a functional relationship between input and output variables. An alternative way to characterize the problem, therefore, is to say that failure to model the underlying mechanisms of performance results in models that are difficult to generalize. This failure to model internal mechanisms has resulted in the development of many good, but limited, HPMs.

Cognitive psychology, in contrast, is primarily concerned with delineating the mechanisms underlying human performance. Cognitive psychology investigates mechanisms involved in perception, memory, and thinking (Collins and Smith, 1988). Cognitive psychology borrows formalisms developed in artificial-intelligence research and uses them to construct computational models of the mind. Such models are often called process models because they depict the process of information flow and transformation.

One problem with most contemporary process models is their lack of sophistication in depicting the information-processing system. Contemporary models usually depict only a limited part of the information-processing system. For example, most models only depict the serial processor operating within human awareness; they do not depict the parallel processor operating outside human awareness. In addition, most models only depict bottom-up data processing; they do not depict top-down, conceptually driven, data processing. Consequently, most contemporary process models are also very limited in the behavior they can effectively portray.

Additionally, contemporary process models are often criticized as not being neurally plausible. The human brain consists of a very large number of units (neurons) organized into a large number of ensembles that work together to give rise to human performance. The ensemble of neurons apparently has multiple feed-forward and feed-backward connections (Damasio, 1989; Edelman, 1989). In contrast, most contemporary models are modeled after a Von Neumann computer architecture in which information is stored remotely in symbolic structures and brought to a central processing unit to be manipulated.

HPP-model research shares the legacy of both engineering psychology and cognitive psychology. Currently, an attempt is being made to develop models that can be used in engineering studies to evaluate design alternatives. The approach, however, is to model the processes and mechanisms which give rise to performance. The goal is to build models sophisticated enough to address the limitations and issues identified by the NRC studies.

The approach taken in this paper is to begin with a discussion of levels of abstraction, then to address issues associated with psychological reality and the difference between psychological constructs and computational mechanisms. Next, the levels at which a model should be specified are described and the cognitive architecture being developed is introduced. Finally, a methodological approach to developing HPP models (the method of successive approximation) is presented and the modeling effort is discussed in terms of the research issues identified by the NRC studies.

SYSTEM LEVELS AND LEVELS OF ANALYSIS

For any system that is being studied, there are multiple perspectives from which to contemplate the system. These varying perspectives are called system levels or, alternatively, levels of analysis. In general, individual system levels are meaningful perspectives of the system under study, where "meaningful" is defined by an individual researcher's requirements. Customarily, individual levels within a given framework describe the system at different levels of abstraction; that is, one level may describe the system from a computational perspective, while another describes the system from a functional perspective.

There is an active discussion in the research literature concerning the concept of levels. Much of the discussion focuses on how many unique levels there are, what they should be called, and whether they are psychologically real. The purpose of discussing system levels herein is not to join in that discussion but rather to frame the current research in terms of existing descriptions. Therefore, the literature on system levels and levels of analysis will be selectively reviewed.[1]

From our perspective, system levels are part of a model or framework specifying how to build HPP models. System levels are independent perspectives which must be specified in order to achieve successful system development. The two proposals reviewed in this paper are the most useful frameworks for designing HPP models.

Marr (1982) proposes a tripartite approach to understanding information-processing systems. The top level in Marr's formulation, the computational theory (CT), defines the system in functional terms. At the CT level, information flow and transformations are described in abstract terms, independent of their realization in a system. The purpose of developing a CT-level theory is to demonstrate the appropriateness and adequacy of the information and information transformations to produce the behaviors to be performed by the system (Marr, 1982). Developing a CT-level specification requires the creation of a functional decomposition of the system which specifies *what* the system does and *why*.

[1] See Anderson (1990) for a more complete discussion of the issues.

Marr's middle level, the representation and algorithm (RA) level, explicitly describes how information is represented and the algorithms that transform the information. At the RA level, information format (or code), content, and organization are specified. Developing a representation and algorithm specification requires describing *how* the system processes information.

The lowest level, the hardware-implementation level, defines how the representations and algorithms are implemented physically in terms of a computational device. At the hardware-implementation level, a specific technology (e.g., biological or computer) is identified to physically *realize* the system.

Marr's research investigated the visual system. He strongly believed that to understand perceptual mechanisms one had to understand the nature of the visual problems those mechanisms solve. Furthermore, Marr believed that distinct visual phenomena should be understood at the appropriate level. Some phenomena are best explained at the hardware/implementation level, while others are best explained at the algorithm level.

Anderson (1990) proposes four levels of analysis for cognitive theories. The highest level is the rational level. At the rational level, the system is understood without reference to or knowledge of internal mechanisms. At this level, an individual provided information on the goals of the system and the knowledge the system possesses can predict the behavior of the system solely by assuming the system will behave in a rational manner.

Anderson's next level is the algorithm level. Anderson describes this level in terms of Newell's physical symbol system hypothesis (Newell, 1980). Symbols and symbol structures are the medium through which the algorithm level is instantiated. Information-processing at the algorithm level produces state changes in working memory. Anderson believes that the distinction between hardware and software in computers is also found in the mind. For Anderson, an algorithm-level specification represents software that can be realized by many different "hardware" implementations.

The next level is the implementation level. The implementation level is an approximation to the lowest level--the biological level. Anderson proposes this level as a construct to help define the computational costs of algorithm-level operations. Anderson states that the implementation level has identifiability problems. He maintains that it is not possible to decide among competing claims at this level (e.g., Are processes serial or parallel? Is there a distinct short-term memory?); rather, this level is useful as a way to set bounds on the possible ways the algorithm level may be realized.

The lowest level is the biological level. It represents the computational substrate of the brain and is similar to Marr's hardware-implementation level. However, Anderson calls it the biological level to emphasis that it refers to the brain.

In comparing the proposals of Marr and Anderson, two concepts should be considered: (1) the issue of *psychological reality* and (2) the study of *cognitive processes* versus *perceptual process*.

Anderson states that he agrees with Marr in principle. However, he offers his own proposal concerning levels of analysis as an attempt to define what can be shown to be psychologically real. Psychologically real mechanisms (Anderson, 1987; 1990) are mechanisms that actually exist in the mind. Anderson argues that only the algorithm and biological levels are psychologically real, and expresses doubt concerning the human ability to explicate the functioning of the mind at the biological level. He proposes the implementation level as an

intermediate representation necessary, principally, to simulate cognition within a computer and to anchor psychological theories.

The central issue of psychological reality is, of course, the black box phenomenon. Basically stated, this phenomenon is the inability to distinguish, on the basis of external appearance, between two black boxes that compute the same input-output functions. Black boxes that compute the same input-output functions are functionally equivalent Turing machines. Distinguishing between two black boxes requires a detailed investigation of the mechanisms within the "boxes." However, a human brain cannot be opened up to allow a detailed study of the mechanisms within it because of moral and legal obligations. Hence, it is very difficult to determine what mechanisms actually underlie the functioning of the mind.[2]

Psychologists use experiments to explore the black box of the mind. Typically, they present subjects with stimuli and measure the responses to explicate the functionality of the mind. Based upon experimental results, it may be proposed that the mind possesses a certain functional process or construct (e.g., short-term memory or attention). The problem, however, as pointed out by Anderson and others (e.g., Barsalou, 1992), is that the plausibility of these processes rests on their ability to explain behavioral data. They are not tied to neurological research and are indistinguishable when compared to other proposals that explain the same behavioral data.

The problem of the black box and psychological reality is exacerbated when one studies "cognitive" mechanisms versus "perceptual" mechanisms. In general, when studying perceptual mechanisms (e.g., vision) there usually are some clearly defined constraints that can help guide and validate research. There is, for example, a limited spectrum of light that can be perceived. Knowing this can help determine the frequencies and intensities that the eye is sensitive to and limits the domain to be investigated. Furthermore, there is often neurophysiological data available from research on other species with very similar mechanisms to help in the validation process. This data allows researchers to correlate physical structure with behavioral function and identify the pathways from an external stimulus to a physiological response (Shatz, 1992).

In contrast, similar constraints are lacking in the study of cognitive mechanisms. While it is generally agreed that individuals make decisions, engage in planning, prioritize the execution of activities, and engage in other types of cognitive activities, there is a distinct lack of agreement on a taxonomy of such activities. In addition, as evidenced by Anderson's article on research methodologies and the 22 replies to the article (Anderson, 1987), there is a lack of agreement on accepted research methods for studying cognitive processes. Some researchers believe that obtaining protocol data is the best approach; others believe that collecting reaction time data is better. Finally, there is inadequate neurophysiological data on the underpinnings of cognitive processes in the brain. There is good data from other species on perceptual mechanisms, but there is no similar data for cognitive processes. Combined, these problems make it very difficult to establish the psychological reality of proposed cognitive mechanisms.

The difficulty of establishing the psychological reality of cognitive processes is reflected in Marr's and Anderson's levels-of-analysis proposals. Marr, who studied perceptual processes, was confident of the psychological reality of the mechanisms he was studying. In contrast, Anderson, who studies cognitive processes, is much less comfortable with the psychological reality of the processes he studies. Thus, Marr proposes three levels of analysis, all of which are

---

[2] There are, of course, some exciting, new physiological measurement techniques being developed to address this problem. Eventually, positron emission tomography (PET) or the newer functional magnetic resonance imaging (MRI) may provide the data needed to distinguish between theories of the mind.

"psychologically real,",while Anderson proposes four levels, some of which are more (or less) psychologically real than others.

PSYCHOLOGICAL CONSTRUCTS, REIFICATION, AND COMPUTATIONAL MECHANISMS

The model designer must be wary of the tendency to *reify* psychological processes; that is, to regard something abstract as if it were a real or concrete entity. Consider, as an example, the theoretical construct, attention. Psychologists invoke attention to explain the process by which humans reduce the information bombarding them. The basic idea is that to perceive any stimuli one must first attend to it. Attention research attempts to explicate the limits of, and processes involved in, human attention.

Reification leads us to refer to attention as if it were a "living spirit" employed as a gatekeeper or searchlight operator in service of the homunculi within the brain. If this sounds far-fetched, consider the ways attentional processes are described. They are "agents" or "daemons"; that is, active processes "scanning incoming information" or "operating perceptual gates."

The issue is that, while there is no doubt that the contents of consciousness change or that these changes follow some form of law-like or predictable patterns, the basis of the mechanism underlying these shifts of content in awareness remains indeterminate. Is there a switching mechanism or something else? If it is a switching mechanism, is there one switching mechanism or several? If changes in consciousness are due to switches, what is being switched? Without careful specification of the underlying mechanism, reification of the psychological constructs can occur, imparting an unearned aura of psychological reality to the construct.

In designing process models, the model developer must be especially careful to design *computational mechanisms* which preserve the same input-output functions discovered by the original experimental studies and not to create a "celestial hierarchy" of computationally under-specified psychological constructs. Many psychological constructs appear useful as a shorthand way of describing some information-processing activity. However, in reality, these constructs may be epiphenomena arising from the interaction of external stimuli and indeterminate internal mechanisms. Further exploration of the traditional concept of attention will help clarify this consequential point.

As discussed by Kahneman and Triesman (1984), there are two dominant research paradigms investigating the limits of attention: the filtering paradigm and the selective-set paradigm. These paradigms differ in the methods and procedures they employ to study attention. The filtering paradigm is distinguished by three features: (1) the subject is exposed simultaneously to relevant and irrelevant stimuli, (2) the relevant stimuli control a relatively complex process of response selection and execution, and (3) the property that distinguishes relevant from irrelevant stimuli is normally a simple physical feature. In the selective-set paradigm, the subject is prepared for a particular stimulus and is instructed to indicate as quickly as possible the detection or recognition of that target. Thus, in the selective-set paradigm the subject is searching for one of several stimuli, whereas in the filtering paradigm the subject is analyzing multiple stimuli.

The filtering and selective-set paradigms were created to study different aspects of attention. The filtering paradigm was developed to study the limits of performance and to measure the extent to which different tasks can be combined without loss. The selective-set paradigm was developed to study the brain's ability to resist distraction, and to establish the locus beyond which relevant and irrelevant stimuli are treated differentially. Research results derived from the filtering paradigm suggest (in general) that the brain is organized as a modular system

by sensory modalities and that interference between stimuli arises chiefly within rather than among the separate, semi-independent subsystems. Research results derived from the selective-set paradigm suggest that the brain has an impressive ability to parallel process multiple stimuli, even within the semi-independent subsystems. These results are obviously contradictory. The filtering paradigm results imply that an HPP model should have a processing bottleneck in each modality; the selective-set results suggest that an HPP model should have parallel processors in each modality. A possible way to resolve this dilemma is to consider whether these two experimental paradigms are studying the same phenomenon.

The concept of attention is a construct; as such, it is a *first-order approximation* to a computational process occurring within the brain. The issue becomes, therefore, a question of whether there is *one* computational mechanism underlying or causing shifts in awareness, or *several.* If there is more than one mechanism or process contributing to "attentional effects" the two research paradigms may be studying different phenomena and the results are no longer conflicting.

This issue of nonidentifiablity strongly affects the way model developers can utilize existing theories and models in creating an HPP model. Contemporary psychological research has resulted in the development of a plethora of potential mental processes (i.e., psychological constructs). These hypothetical processes are often described by analytic models. Analytic models are *second-order approximations* to computational mechanisms. They mathematically model an input-output function; however, they say nothing about the mechanisms producing the behavior seen in the original experiments.

To help make these points clearer, consider Figure 1. Arrow **A** represents input stimulation from a selective-set-type experiment. Arrow **B** represents input stimulation from a filtering-type experiment. Arrow **C** represents a response output from a selective-set experiment. Arrow **D** represents a response output from a filtering experiment. Function $f(\mathbf{A})$ represents an analytic model that preserves the input-output function of the mapping **A input - C output**. Function $f(\mathbf{B})$ represents an analytic model that preserves the input-output function of the mapping **B input - D output**. Function $f(\mathbf{X})$ represents an analytic model which preserves both the input-output functions of mapping **A input - C output and B input - D output**. The box titled "Brain" represents, obviously, a brain. The boxes and circles represent hypothetical computational mechanisms within the brain.

The question is: How do we know what set of mechanisms an experiment is probing? To an outside observer the brain is a "black box." It is an indeterminate system; very little is known about the mechanisms actually processing information. A typical experiment probes the brain by presenting a stimulus set and recording output. Based on the experimental results, a construct (e.g., attention) is proposed to help elucidate and communicate the findings. An analytic model defining a function producing the same mapping of input-to-output data may also be developed. All this is very useful science; however, what has been learned about the mechanisms that actually process information in the brain? The answer is very little. In the absence of neurophysiological data, it cannot be determined whether the set of brain mechanisms being probed in different experiments are the same or different. (The nonidentifiablity problem is well-known to scientists who study attention. Attention researchers are in the forefront of those employing the new physiological measurement techniques [e.g., PET and MRI] in psychological studies.)[3]

[3] See Allport (1993) for a discussion of problems for theories of attention.

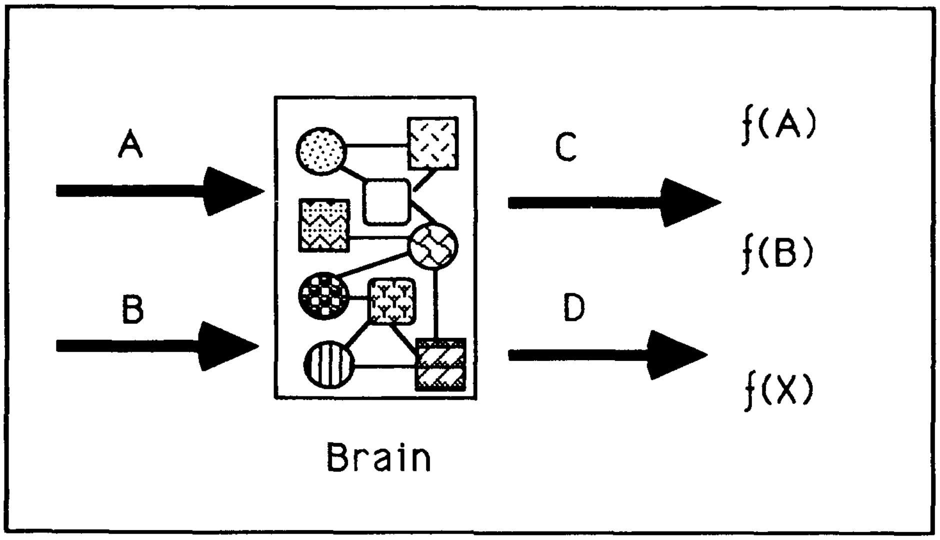

Figure 1
One Approach to Research

This emphasis within psychology of proposing constructs--which may or may not be real--and describing these constructs as analytic models--which say nothing about the computational mechanisms underlying these proposed constructs--has resulted in the development of a profusion of micromodels of information processing that cannot be combined to create a comprehensive information-processing model (Elklind et al., 1989).

SYSTEM LEVELS AND HUMAN PERFORMANCE PROCESS MODELS

This effort proposes a method for developing HPP models that is adapted from a traditional systems-engineering approach for developing systems. The technique differs from a traditional systems-engineering approach in that HPP models must be specified at multiple levels of abstraction. In this section, the levels at which HPP models must be specified are described. In general, we agree with Marr's formulation of the issues; however, we extend Marr's concept of the CT level to include architectural issues. Furthermore, to be clear about issues of psychological reality, what is claimed to be a model of the mind is distinguished from what is a computer implementation. Therefore, Anderson's concept of an implementation level has also been adapted. Thus, it is proposed that an HPP model should be specified at four levels of abstraction, two of which are psychologically real and two of which are not.

Computational-Theory Level

The first level to be specified is the CT level. The CT level in this research is both similar to and different from Marr's conceptualization. It is similar in that at the CT level one must specify the functional behavior of the system and explain how this behavior allows the system to accomplish its goals or functions. To achieve this specification, the tasks the system must perform are analyzed into an information-processing perspective which defines the mapping of information from one state (or stage) to another to demonstrate the adequacy and appropriateness of the transformations for task accomplishment (Marr, 1982). This analysis is performed abstractly, that is, without reference to how these transformations will be implemented in computational mechanisms.

Marr's proposal is extended under this research effort by requiring that, in addition to specifying the functional behavior of the system, one must also specify a cognitive architecture as part of the CT description of the system. A cognitive architecture establishes the tenets that define possible relationships among knowledge sources (i.e., the functional processes that transform information). The cognitive architecture defines a design space of possible topologies for knowledge sources and formalizes the functional interdependencies between the modules that comprise the mind.

Specification of a cognitive architecture is required, in part, to acknowledge that one is not specifying a general class of information-processing systems but rather the human class of information-processing systems. The human brain is the product of evolution. It has evolved functional capabilities to adapt to, or solve, information-processing problems presented by the environment. Individuals, in general, share the same cognitive adaptations or functional processes needed to process information. The cognitive architecture defines or captures the similarities in functional processes across individuals at a functional level of abstraction.

The functional level of abstraction, as defined by the cognitive architecture, is the most appropriate level to define similarities between individuals. Consider the ways in which individuals of the class of human beings can be said to be the same. A human brain consists of approximately 100 billion cells, which collectively have 100 trillion connections. Neurophysiological research has shown that individual brains are individually wired. The wiring of individual neurons in the brain is not genetically programmed; rather, it has been demonstrated that neural wiring is dependent upon environmental stimulation. No two human beings--not even genetic twins--experience exactly the same environmental stimulation. Hence, no two brains are identical in terms of their neural wiring. In addition, research has shown that the neural wiring is constantly undergoing modification. New synapses are formed and some existing synapses degenerate; thus, the wiring of individual brains changes over time.

Additionally, experimental psychological research has shown that individuals differ in terms of cognitive capabilities. For example, research has shown that there is cognitive variation between the sexes and between left- and right-handers (Kimura, 1992). Men (generally) are better at spatial tasks and mathematical reasoning while women (generally) are better at matching tasks and have greater verbal fluency. In addition, right- and left-handers differ in the functional organization of their brains. These differences include both differences in the how the brain processes spatial information and other cognitive processes, such as speech understanding.

Although there are differences in the ways individual brains are wired and the functional skills and topographical layout of functions within individual brains, in some sense all humans process information in (approximately) the same way. Humans represent a class of information-processing systems; the cognitive architecture defines this class at a functional level of abstraction.

In addition to the conceptual need for a cognitive architecture, there is a practical reason for specifying a cognitive architecture: it creates a set of constraints on the range of possible computational mechanisms which comprise the human information-processing system. The introduction of this paper discussed a NRC study (Elkind et al., 1989) which reviewed HPMs to determine if a set of models existed to support the development of an HPM for a CAE facility. The review was not limited to process models but included other kinds of models (e.g., analytic models, qualitative models, etc.) as well. The study, which reviewed several hundred models, found that available models would not support the goal of providing a complete simulation of visual and cognitive processes.

A key finding of the NRC study was the lack of a satisfactory architecture that could provide a framework to integrate existing micromodels of information processing. This lack of

architecture was particularly evident in regard to cognitive processes. The study points out that, for cognition, an architecture that allows association of information-processing micromodels to stages of processing is not available. Furthermore, cognitive aspects of the information-processing system interact in such a complex fashion that it is difficult to separate processing into either distinct functional processes or discrete subcomponent models.

An explicit cognitive architecture is one way to address this problem. The cognitive architecture provides a framework into which micromodels can be integrated. As a framework, it represents a plausible *design space* in which information-processing models can be developed and compared. As a design space, the tenets of the cognitive architecture provide a new set of constraints with which to compare and evaluate models. This implies that, in addition to being scientifically sound, individual micromodels must also be readily mappable into the design space defined by the architecture. The ability (or inability) to map micromodels into a cognitive architecture helps define sets of models that share pretheoretical assumptions (Lachman et al., 1979); hence, it can serve as the foundation for developing an integrated and comprehensive HPP model.

Because of the importance of the concept of cognitive architecture and because the use of the term in the current effort is slightly different from the way other authors have employed this concept, an example of a cognitive architecture is provided in the following paragraphs to help clarify the concept. The example comes from the current effort to develop advanced HPP models.

The holon cognitive architecture (Young, 1992) is based upon Koestler's (1967, 1976) theory about the hierarchic behavior of organizations. The holon architecture postulates that the human information-processing system is a set of modules, called holons, arranged into a hierarchy, called a holarchy.

Individual holons (modules) behave as if they were quasi-autonomous wholes. Each holon displays its own asynchronous timing basis for patterns of activity. In addition, each holon has its own internal representation (working memory) of the environment. The "environment" for a holon may be external or internal (to the system), or both. Holons represent only the portion of the environment that is relevant to them in performing their function. Further, individual holons can represent information at different levels of abstraction.

A holon's behavior is governed by fixed rules, collectively called the holon's canon. This canon is executed through flexible strategies and execution is dependent upon the its environment. Holons can potentially generate a wide range of behaviors from a limited instruction set (canon) due to the diversity in the environment; thus, holons have great behavioral flexibility.

Holons communicate using three types of message-passing: hierarchical, broadcast, and point-to-point. Hierarchical message paths define parent/child relationships. Broadcast messages go from one holon to all others. Point-to-point messages go from one holon to another holon (or set of holons).

Holons are linked together in a hierarchy called a holarchy (Koestler, 1976). The number of levels in a holarchy is the holarchy's depth; the number of holons on a given level is the holarchy's span. Figure 2 is an input holarchy with a depth of two and a span (at the lowest level) of four. Levels within a holarchy demarcate processing points where the level of abstraction of information changes. Typically, information comes in at one level of abstraction, and through processing within the holon, a new information product, usually at a different level of abstraction, is generated.

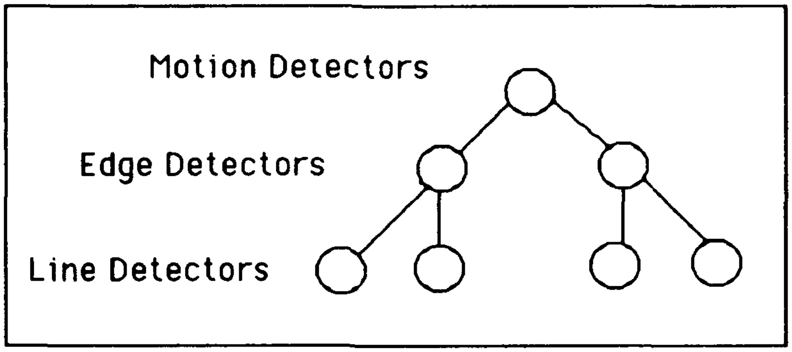

Figure 2
Visual Input Holarchy

There can be different classes of holarchies. The two most common classes are input holarchies and output holarchies. Input holarchies convert complex input patterns into symbolic representations. Each level of an input holarchy acts as a filter or classifier, identifying the input signal as some higher-order construct. The abstracted knowledge of that construct is then passed up the holarchy.

Output holarchies work inversely: an input holarchy abstracts information whereas an output holarchy elaborates or embellishes it. For example, a simple signal called the trigger releaser can cause the output holarchy to perform complex actions. The releaser concept is analogous to the forming of an intention. Once the "intention" activates the holarchy, holons on each successive lower level of the output holarchy further define and specify the action being undertaken.

Specific branches are established in a holarchy to either analyze or define (depending on whether it is an input or output branch) specific dimensions of the signal. For example, recent research on the visual system (Zeki, 1992) suggests there are separate visual pathways for analyzing the movement, color, and shape of visual stimuli. These separate pathways would be represented as separate subbranches within a visual branch of a holarchy.

In summary, the holon architecture provides a set of constraints and a framework within which to develop HIP models. It also provides an architecture into which micromodels can be mapped. Continuing with the visual model introduced in the previous paragraph, micromodels representing various stages of analysis of movement, color, and shape can be mapped into individual holons within the proper subbranches of the model. These micromodels are integrated by providing the output of one micromodel to other models needing this output as input via message-passing.

As it is currently formulated, the holon cognitive architecture is underspecified. The purpose of developing a CT specification is to specify the functionality of the system at an abstract level. The above formulation of the holon architecture does not specify required functionality; rather, it merely delineates a design space. The problem with specifying functionality is that there is not an agreed upon taxonomy of the mental processes of the mind. In the method of successive approximation, a methodology for determining mental requirements is proposed. Therefore, the issue of defining functional requirements for mental processes will be discussed further in the section on requirements identification.

Representation and Algorithm Level

The RA level provides the specifics of how information is processed and transformed. The RA specification is a design-stance (Dennett, 1978) view of the system. It defines a model of how a specific instance (person) of the class of human information-processing systems actually processes information. The RA specification depicts information-processing components at a symbolic level of abstraction and describes how information is transformed.

In an RA-level specification, topography of the model must be specified; thus the functional purposes of individual holons (modules) must be defined and the holon communication paths must be specified. In addition, for each holon, the information content, format (or code), and organization (data structures) must be specified, thereby requiring the specification of the aspects of the environment that will be stored in each holon's working memory. Finally, the algorithms or heuristics that will manipulate the information within a holon must be specified.

The RA-level specification is psychologically real. It is a hypothesis about how a specific individual processes information. Individual RA-level specifications are varied to reflect differences in how individuals process information. The assignment of functionality to individual holons, the communication paths between holons, and/or the processing specifics of individual holons are varied to capture individual differences. Individual differences depicted in this manner include differences in skill levels and decision-making strategies, and changes in task performance because of learning.

Implementation Level

The implementation-level specification casts the psychological model defined in the RA specification into computational terms. As proposed by Anderson (1990), this level is not psychologically or physiologically real. It is an approximation employed to simulate a biological system within a computer. Thus, the implementation level demarcates the difference in designing an HPP model between psychologically real and not real. The CT and RA specifications are psychologically real; they represent hypotheses about the psychological functioning of the human mind. In contrast, the implementation and computer levels are not "real"; they are techniques utilized to simulate the psychological models within a computer.

Implementation-level specification delineates the choice of data structures and data-manipulating techniques (inferencing techniques, algorithms, etc.) to be employed to realize the CT and RA specifications. Choice of an actual implementation-level specification is based, in part, on additional requirements best realized at the implementation level. For example, the Armstrong Laboratory employs HPP models as "team players" in real-time, human-in-the-loop simulations and to conduct cognitive science research by comparing the performance of alternative HPP models to human-in-the-loop operators. These applications require that the HPP model implementation have real-time simulation capability and be highly modular.

The constructs required to support code modularity and the speed required to support real-time simulation are usually not found in the same language. Program languages that are highly modular (e.g., Smalltalk or Lisp) are often interpreted and, hence, fairly slow. Conversely, some "faster" languages are not highly modular, nor do they readily support a wide range of data structures or inferencing techniques. At the implementation level, the designer must tradeoff these issues to develop the best implementation strategy to realize the RA-level specification within a computer.

Computer-Architecture Level

The last level to be specified is the computer-architecture level. Computer-architecture specification is the choice of computer platform. Choices are made among different computer architectures (e.g., serial, parallel, neural-net) and among different processors within an architecture. In choosing a computer architecture, one compares the requirements of the implementation specification to the capabilities of various architectures to determine the best match.

One possibility not yet explored is building HPP models that employ multiple classes of computer architectures. For example, one might want to use a traditional architecture for knowledge-based processing and a neural-net processor for visual pattern recognition. This requirement could develop as a result of an inability to design an efficient implementation-level specification (e.g., one that could support real-time visual processing) using just one computer architecture. The holon formalism readily supports this type of implementation. In principle, each holon can employ a different type of computer to achieve its processing. In a mixed computer configuration, holon message-passing would occur over a "bus" linking the different processors.

THE METHOD OF SUCCESSIVE APPROXIMATION

To support the development of predictive HPP models, a new approach to creating HPP models, called *the method of successive approximation*, is being developed. The central idea of this method is to incrementally develop and test HPP models. The method of successive approximation begins with the development of an initial HPP model that is comprehensive in the depiction of the information-process system, although limited in the range of behaviors it can accurately reproduce. Over time, the behavioral repertoire is progressively extended through additional development cycles.

There are two key aspects to this approach: one technical, one conceptual. On the technical side, the ability of object-oriented programming techniques to create and maintain libraries of HPP model components (i.e., subcomponent models) at multiple levels of resolution and representing alternative theoretical perspectives is being investigated. On the conceptual side, conventional system development methods are being applied to the creation of HPP models. This methodology provides the basis for incrementally developing the HPP models.

Object-Oriented Programming

Object-oriented programming (Bobrow & Stefik, 1986) is the technical enabling technology for the method of successive approximation. Objects are self-contained program elements which perform computations and maintain a local state. In addition, objects interact with other objects (communicate) by sending messages. For example, one object might send a message to another object requesting that some computation be performed. Objects support polymorphism; that is, different classes or types of objects can respond to the same message-passing protocol. In addition, objects are encapsulated; information within the object is "hidden" from the rest of the program. Information entering or leaving the object goes through a port which is a specialized method. Furthermore, object-oriented programming supports code reusability. Together, polymorphism and information encapsulation create a programming environment in which major parts of the code can be removed, replaced, or changed with minimal recoding of the rest of the program.

Object-oriented programming is a natural way to represent a holarchy. Individual holons become individual objects. Since each holon is the equivalent of a separate computer process, creating holons as objects is a natural way to represent them. In addition, the hierarchical aspect

of the holarchy is achieved by programming convention through the use of parent and child message destinations. Each object has one parent and (potentially) several children. Furthermore, information encapsulation allows objects to maintain a separate working memory for each holon. Finally, holons and objects both communicate via message-passing, so there is a natural correspondence between the two.

Object-oriented programming supports model development at different levels of *resolution*. Resolution refers to the level of detail at which model subcomponents are specified. For example, in developing a visual subcomponent for an HPP model, one can specify the constituent pieces (e.g., two retinas, intermediate processing layers, integration centers, cognitive center) in great detail or combine several component pieces into one knowledge source (processing center). A model for which the constituent pieces are specified in great detail has greater resolution than one for which the pieces are combined into one knowledge source.

There are two types of resolution: aggregation and abstraction (Fishwick, 1986). In aggregation, detailed knowledge about the process to be aggregated is available and is the basis of aggregation. The aggregated knowledge source retains all the functionality of the aggregated subprocesses. Conversely, in abstraction, detailed knowledge about the subcomponent processes is not available, and the combined knowledge source represents what is known about higher-level system characteristics.

In the method of successive approximation, a complete model is developed during each iteration. Initial models, however, have some components specified at low levels of resolution. Through successive increments of development, the behavioral range of the model is extended by increasing the resolution of the model. Model resolution can be extended both "horizontally" and "vertically." Increasing resolution horizontally increases the number of psychological processes being modeled; increasing resolution vertically increases the level of detail at which these processes are modeled.

The resolution of a holon-based HPP model can be extended in two ways. First, a holon at a high level of resolution may be broken up into several new holons performing the same functions as the original. The additional holons add detail by further specifying the workings of the processes accomplished by the original holon. Breaking up a holon requires modification of the message pathways of the holons which communicated with the broken-up holon. Additionally, the internal workings or mechanism within a holon can be further detailed to increase the resolution of the model. A simple function could, for example, be replaced with a more elaborate computational mechanism. This type of modification does not require any changes to other aspects of the code. Object-oriented programming is the underlying technology that makes changes in model resolution achievable with an acceptable amount of effort.

Systems-Engineering Methods

The conceptual enabling technology for the method of successive approximation is a systems-engineering methodology. Systems-engineering methods provide a model of how system development should proceed, a set of procedures to follow in developing the system, and evaluative criteria with which to evaluate progress (Svoboda, 1990). Incremental development of HPP models requires a methodology to ensure the completeness of the HPP models developed and to provide support to HPP model development across increments of development.

In general, all systems-engineering techniques follow the same basic steps of requirements identification, design, development, test, and operation; however, techniques differ as to when products are developed, how risks are managed, and the role of prototyping in system development. The specific approach of interest in the current effort is called the incremental

development approach (see Figure 3). Incremental development consists of a series of development cycles, each resulting in a usable product.

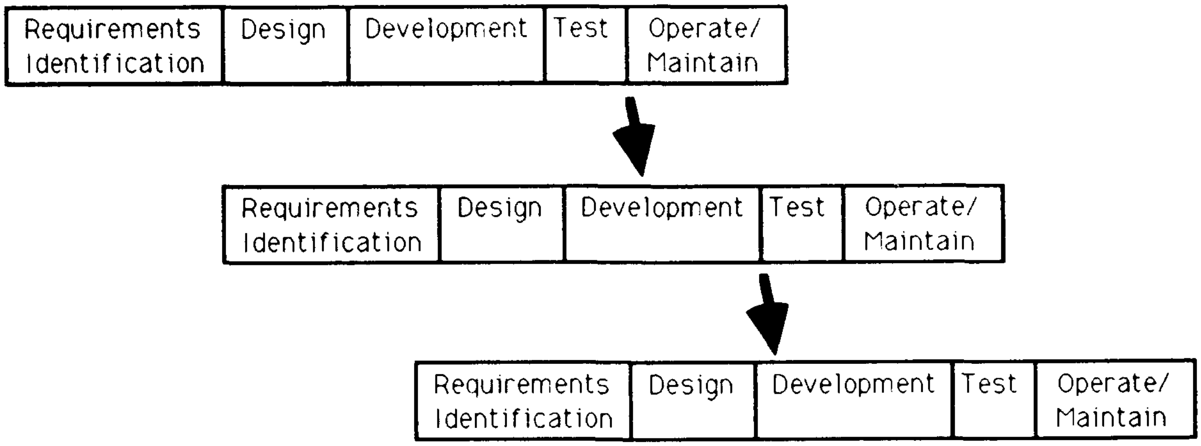

Figure 3
Incremental Development Process

In the method of successive approximation, the incremental development methodology is adapted to the development of HPP models. Developing HPP models is more complex than developing a conventional system due to the requirement to *depict and simulate* psychological processes. During each development cycle, a complete HPP model is developed, tested, and validated. Each development cycle focuses on reproducing one aspect of human performance behavior important to operability analysis. For example, an initial cycle might focus on modeling skill-level differences between novices and experts. A later cycle might focus on modeling complex decision-making. Each successive cycle extends or refines the initial model.

Requirements Identification

The first step in designing an HPP model is requirements identification. In a conventional system development process, requirements are the functions the system must be capable of performing. Similarly, when building an HPP model, requirements are the behaviors the HPP model must be capable of performing. In the current research, HPP models are being employed as part of a system modeling test-bed (Young, 1992); therefore, the behaviors to be modeled are those required to perform the mission of the system being evaluated. These behaviors are identified through the development of a functional task decomposition which translates mission requirements into a set of discrete tasks capable of accomplishing the mission. Furthermore, when identifying behaviors the HPP model is to perform, it is important to identify the environmental conditions in which the model will operate and the representative scenarios in which the missions will be performed.

The functional task decomposition is best represented as a multilevel goal hierarchy (Adams et al., 1991) in which individual goals and the tasks required to meet those goals are represented on different branches of the hierarchy. Representing tasks as a multilevel goal hierarchy is an excellent way to depict the logical and functional interdependence of tasks. In addition, a multilevel goal hierarchy represents tasks independent of the activities required to execute the tasks, thus allowing tasks to be easily mapped against multiple system design configurations.

Deutsch and colleagues (1993) have proposed that for each task in the hierarchy the following eight items be documented: (1) steps required to perform the task; (2) input required to perform the task (including type, external source, and modality received); (3) output generated by the task (including type, destination, and modality); (4) decisions required to accomplish the task; (5) processing required to make the decision; (6) local data required to make the decision; (7) domain-specific knowledge required for task accomplishment; and (8) control knowledge/strategies. In addition, a trigger stimulus for task activation should be identified and documented.

The requirements analysis must also ascertain the environmental conditions under which the system and HPP models will be employed. Typical environmental conditions of interest include lighting, thermal, and noise conditions. Knowledge of environmental conditions is used to help determine the level of resolution required in the model.

Finally, the requirements analysis should develop scenarios representative of the conditions in which the total system--computer-based control system and operators--will be employed. Ideally, these scenarios should represent a variety of workload conditions (including both normal and emergency operating conditions) to provide a full spectrum of system-employment conditions. Scenarios are best listed as a time-line analysis. For each task to be performed in the scenario, data is collected on task priority, the time frame in which the task can be performed, the time required to complete the task, and the demand level the task places on the operator.

The output of the requirements-identification stage is a functional decomposition of the tasks to be performed by the HPP model represented as a multitask goal hierarchy, a listing of pertinent environmental conditions, and a set of representative scenarios depicted as a time-line analysis.

Design Stage

In the design stage, the developer designs an HPP model with sufficient psychological capability to perform the required tasks. The design process in the method of successive approximation differs significantly from a conventional approach to design because of the need to specify the HPP model as both a psychologically real representation of the mechanisms underlying human performance and as a computer-implementable simulation. Achieving this design goal requires three phases of design work. In the first phase, the CT specification is developed; in the second phase, the RA specification is developed; and in the third phase, the implementation specification is developed.

However, before describing the proposed design approach, the differences between analytic and HPP models should be reviewed and the role of these models in the proposed method should be discussed. Process models are being developed in the current effort. These are engineering models that attempt to accurately depict the computational mechanisms underlying human performance. In contrast, analytic models are mathematical models that preserve or depict a function discovered by experimental psychological research. Analytic models are second-order approximations to computational processes (the first-order approximation is the construct under study). Every HPP models is actually a *combination* of both classes of models.

The key issue is resolution. HPP models do not normally depict individual neurons within the brain. Therefore, they are not true or complete process models. Rather, they are hybrid models consisting of some process-model subcomponents and some analytic-model subcomponents. The model changes from a *process model* to an *analytic model* at the level at which the modeling of "real" computational processes stops. That is, the lowest level of

holarchy is an *abstracted* analytic model of neural processes. (Given what little is known about the neurobiological basis of behavior, the lowest level is, in reality, just a plausible approximation to neural processing.)

Furthermore, the ratio of process model to analytic model changes through increments of development. Initial models consist of a higher ratio of analytic subcomponent models than later models. The method of successive approximation incrementally replaces analytic-subcomponent models with process-subcomponent models. However, it is always true that at the lowest subcomponent level depicted the process model becomes an analytic model.

Computational-Theory Specification

The goal in developing a CT-level specification is to identify the functional processes the model must embody and the level of resolution at which these processes need to be modeled. Developing the CT-level specification is an iterative process (see Figure 4) of mapping the set of behaviors identified in the functional task decomposition onto a set of functional mechanisms (or holons) which produce the behavior (given an appropriate input set). This mapping process is aided by the use of psychological constructs as intermediate representations. These psychological constructs represent hypothetical functional capabilities that the model embodies. Furthermore, during the development of a CT-level specification, the cognitive architecture is employed to constrain or bound the potential forms the computational mechanisms can take.

The specification process begins with the model designer choosing one of the behaviors identified in the functional-task decomposition. This behavior is analyzed to define a set of functional processes required by the model to produce the behavior. Potential processes include perceptual, cognitive, and motor abilities. Potential perceptual processes include capabilities such as scanning and directed eye movements. Potential cognitive processes include capabilities such as decision-making, task priortization, task scheduling, and categorization. Potential motor processes include capabilities such as planning and controlling body movements. Once an initial taxonomy is defined for the first behavior, it is refined through analysis of the representative scenarios and environmental conditions.

Analysis of the environmental conditions helps determine the level of resolution needed in a subcomponent model and can identify additional required functional processes. For example, if an analysis of environmental conditions determines that the operator will employ the system in a well-lighted environment (from a human-factors perspective), this obviates the need to specify the visual system at a high level of resolution (because it would be safe to assume the operator could see all visual stimuli). Alternatively, if an analysis of the environmental conditions determined that the system would be employed in very hot steamy conditions, a mechanism to produce fatigue effects would have to be incorporated into the model.

Notably, there is not a generally accepted taxonomy of psychological functions (processes) for human performance. While most researchers agree that humans can recognize situations, make decisions, prioritize and schedule tasks, and so forth, there is no agreement as to the *functional substrate* needed to perform these tasks. Are there general purpose psychological processes or very specific psychological processes? Creating a CT specification *is not* simply reviewing the behavior to be performed by the model then looking up in a task taxonomy the appropriate psychological processes which need to be incorporated into the model to support task performance. Rather, developing a CT specification requires that one make an explicit proposal about the psychological processes found in the mind.

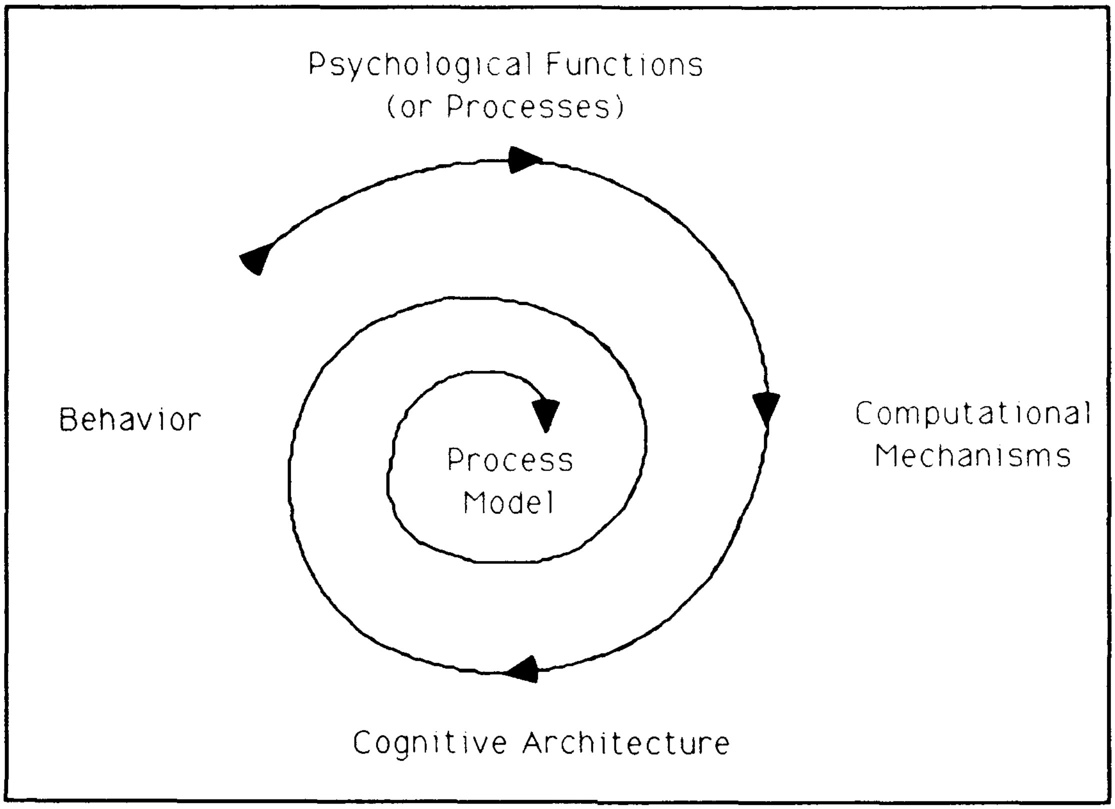

Figure 4
Computation Theory Design Process

The next step in developing a CT-level specification is to define model elements and subelements, and assign psychological functions to them. Typically, identified functions are assigned to holons (given the design space being explored). For each holon, the functional purpose of the computation, description of the computation, the input and output, and working memory requirements are specified. This information, for the most part, is available in the functional task decomposition. At this point, one does not specify the algorithms (or heuristics), the intermediate states of the computation, or the message paths used to the holon; these parameters are defined during the RA-level specification phase. The goal during CT specification is to define the functional requirements of the model rather than a detailed process model.

There is not necessarily a one-to-one mapping of functions to holons; individual holons may perform multiple functions. Furthermore, some functional processes may be mapped to more than one holon. In particular, processes that are modeled at a high level of resolution are likely to be mapped to a single holon.

A key factor influencing the level of resolution at which a process is to be represented is the requirement of the study in which the model is to be used. If, for example, the HPP model is to be used to investigate how individuals of varying skill levels perform using a new command and control system, the model developer depicts those processes believed to underlie relevant skill-level differences at a greater resolution.

In the next step, architectural constraining, the model is examined to determine if it violates constraints imposed by the cognitive architecture. The primary holon architectural tenet constraining the model is the requirement that separate modalities (e.g., vision) in the model be represented by separate branches. In addition, during this step it is important to factor into the model knowledge derived from the cognitive neuroscience literature on the modularity of the mind. The functionality of the model being proposed should be in accordance with what is known from studies on patients with lesions, as well as PET and MRI studies investigating brain processes.

This completes one iteration through the spiral process of developing a CT-level specification. During the next iteration, a new behavior is identified and the process is repeated. However, this time through the spiral, identified functional processes are compared to the list of previously identified processes and new ones are added only as required. In addition, new holons are added only when there is not one available with the right functionality or when a holon's functionality cannot be adapted to meet the additional requirements. An important design goal is to develop a parsimonious taxonomy of processes and holons. Achieving this goal often requires revisions to the existing sets of psychological processes and holons. This process continues until all behavior is accounted for in the CT specification.

The CT specification is next evaluated against the representative scenarios to ensure that the model has all the necessary functionality to perform the mission of the system. The evaluation is conducted as a "walk through" during which the model developer compares the requirements of the scenarios to the functionality of the CT specification to determine if additional functionality is required. An example of additional functionality that might only be found through a review of the scenarios is the requirement that the model be capable of multitasking behavior. A functional task decomposition based on a multilevel goal hierarchy would not contain this information.

The development of a set of holons that contains the necessary functionality to generate the behavior needed to execute the test scenarios concludes the initial CT-specification phase. This specification provides a functional view of the model to be developed. Information flow and transformation have been defined, and functional components (such as input/output devices, processors, and memory storage subsystems) have been identified in terms of their psychological functionality.

Representation and Algorithm-Level Specification

The next step in the design process is to define the functionality of the model in terms of concrete computational processes (albeit, from a psychological perspective). For holon-based models, this process consists of defining message paths between holons and specifying in computational terms how the holons perform their functions. For each holon, inferencing (algorithms or heuristics) techniques, information format, content, and organization are specified.

How does one choose or define a set of computation mechanisms? Ideally, one would directly use the results of years of experimental research, theorizing, and model development to create an HPP model. One would take the list of required functional processes (e.g., visual scanning, short-term memory, etc.) identified in the CT specification and look up appropriate micromodels that would provide the necessary computational processes required for the RA specification. This approach, however, does not work.

Most psychologists adopt an information-processing perspective of the mind. In an information-processing perspective, information is processed and transformed through a series of stages. One issue identified by one of the NRC studies (Elkind et al., 1989) is that there is not an agreed upon set of information-processing stages; in addition, discordance increases as one

moves from peripheral perceptual processes to central cognitive processes. While there is some agreement concerning the required stages of information processing for perceptual processing, there is little agreement concerning cognitive stages of information processing. Furthermore, even for sequences of stages in which there is some agreement as to what the distinct stages are, there are often several competing models of how information is processed at each stage.

Competing models differ in terms of how the information is processed and represented and/or in terms of the required input and output at a given stage. Variations in information processing typically focus on strategies for processing information (e.g., employing an algorithmic or heuristic search) or on the specifics of the inferencing techniques (which specific algorithm or what type of heuristic search). Variations in information representation typically focus on the format, content, and organization of a representation (Kosslyn, 1984). Differences in input and output for a stage may reflect differences in whether a micromodel expects a summation of a spreading activation vector as an input (or provides one as an output vector), or whether an algorithm produces a name or a description of a class and its potential interactions as an output (or requires it as an input) (Kosslyn, 1984).

These model differences are a snapshot of the theoretical differences being debated by scientists. They reflect fundamental differences in beliefs concerning the way information is represented and employed in the mind. Furthermore, most scientists only study a very limited aspect or component of the information-processing system. Hence, they do not think about nor attempt to integrate their research with other researchers who are working on different stages or components of the information-processing system.

The problem for the HPP model developer is how to utilize the body of existing psychological knowledge, as codified in the micromodels of human performance, to create an HPP model based upon the holon cognitive architecture. To create an HPP model, the model developer must reformulate the micromodels and map them into the holon architecture (which provides additional constraints), state them in computational terms (as required by a process model), develop new models to fill in the gaps in information-processing stages or in available models of psychological functionality, and define all the linkages between models.

Obviously, this is not an easy process. To begin with, there are a very large number of existing micromodels (e.g., one of the NRC studies reviewed over three hundred models); however, these models can be sorted into groups based upon the pretheoretical assumptions and intellectual antecedents (Lachman et al., 1979) underlying the models.[4] Examples of contemporary pretheorectical ideas and intellectual antecedents include assumptions that the mind is or is not modular and the mind can or cannot be conceived as a physical symbol system. It is likely that the vast array of existing incompatible models is sortable into smaller groups of reasonably compatible models. If these smaller groups exist, models with similar pretheorectical ideas and intellectual antecedents can be selected.

The current research takes an incremental approach to model development; this approach allows the development of an HPP model for which some subcomponents of the model are depicted at a low level of resolution. Thus, development can be initially focused on key psychological processes (i.e., processes for which there are good models). This approach mitigates the need to resolve all the issues associated with micromodel incompatibility at once. A small subset of micromodels can be worked with in each phase, thereby reducing the amount of work required in any specific phase.

---

[4] Pretheoretical ideas are the conceptualizations of reality held by scientists concerning their subject of study. Intellectual antecedents are the historic or contemporary perspectives that scientists bring to their work. Often these ideas are borrowed from other scientific paradigms for their explanatory value.

Because of the incremental nature of the process proposed in the current research, the model developer must thoroughly document the design. The RA specification is a theory of how the mind processes information. To advance the scientific enterprise, the specification must be stated as a testable psychological model. The theories/models incorporated into the HPP model must be described along with the assumptions and modifications that were required to map these theories into the holon cognitive architecture. This documentation is critical for both testing the model during this incremental development cycle and extending the model in later cycles.

What the documentation should consist of or how it should be organized has not yet been fully defined. For a holon-class model, the interfaces between the holons clearly need to be documented. This information is critical to support changes in model resolution. It is also important to define and document anticipated model behaviors. This information should include the molar behaviors of the aggregate model as well the microbehavior of the subcomponent models. Documentation should identify model parameters and the distributions which define acceptable and unacceptable model behavior. For molar model behavior, this could, for example, include parameters such as time to complete a task and error rates for task performance. For micromodel behaviors, this could include modality bandwidth limits and time to shift attention. Identification of this information is critical to the subsequent testing of the HPP model.

In defining the RA specification, a model developer attempts to create a parsimonious set of computational processes that will give rise to the psychological functionality required by the model. The process is not easy. The model developer can adapt some existing models of subcomponent processes; however, for the most part, a great deal of original design work is needed. The utilization of variable resolution on select model subcomponents, coupled with the knowledge that these underspecified components can be addressed in the next increment of the development process, makes the process workable.

Implementation Design

During implementation design, the model developer takes the psychological model defined by the RA specification and develops an implementation specification defining how the model will be simulated within a computer. The developer chooses a programming language then defines data structures, inferencing techniques, and other programming conventions needed to simulate the model in a computer.

An implementation architecture (Young, 1992) is being created to support HPP model research. An implementation architecture is similar to an application program for drafting, accounting, or data base generation in that it provides a programming language and specialized tools for an application. For the current research, the tools are specialized to support building, running, and analyzing HPP models. The implementation architecture, called the Operator Model Architecture (OMAR), is being developed through a contract to BBN Systems and Technologies (Feehrer &Deutsch, 1993).

The programming language underlying OMAR is the Common Lisp Object System (CLOS). Specialized tools include a frame language with a graphical editing environment, a procedure language with a browser and graphical editor, an interactive real-time execution environment, and an interactive post-run analysis capability.

OMAR is the "programming language" being used to create the HPP models. Hence, in the implementation design phase, the HPP model is designed to take advantage of OMAR's functionality. The developer translates the psychological model depicted in the RA specification into programming conventions supported by OMAR. If OMAR is lacking some necessary

functionality, the model developer can specify new capabilities to be developed through the programming language (which will be created during the software-development phase).

Documentation is also critical during the implementation phase of the design. Since both OMAR and CLOS are inherently object-oriented, all software specifications are in an object-oriented format (Coad & Yourdon, 1990). In addition, a new hypermedia tool is being developed to produce better software documentation.

The output of the implementation-design phase is a specification that can be used by computer programmers to create the model. In addition, this specification provides documentation to support later increments of development.

Software Development

In the method of successive approximation, software development is the process of instantiating the implementation specification into OMAR. The frame language in OMAR provides the ability to define agents and objects. It also provides the link to message-passing and other generic functions in CLOS. Furthermore, the frame language provides tools to create data structures and inferencing mechanisms. The procedural language provides procedure definitions for data-flow networks which instantiate holon functionality. In addition, it has a rule-based language component to support modeling of complex decision-making behavior and a graphical editor to support model setup. Finally, OMAR's interactive run-time component provides tools to control HPP model execution, a filter recorder to capture the audit trial of a simulation run, a filter trace to provide on-line insight into agent/holon performance (as well as debugging support), and a time-line display with the ability to interactively pursue agent behavior forward and backward in time.

The process is very similar to the standard software-development cycle. A programmer starts by defining/developing lower-level objects, progressively works up to developing "higher-level" component modules (objects), and eventually develops the complete HPP model. The programmer is aided in this process through the availability of the special tools (e.g., frame and procedural languages and interactive run-time support tools) in the implementation architecture which reduces development time.

Testing

In developing HPP models, two dimensions of the software must be tested. The first dimension is the functionality of the software, per se. Does the software meet specifications? Does it work as expected? Conventional software testing and evaluation techniques are used to test the software. In general terms, software testing begins with the lowest-level components and proceeds upwards, testing larger- and larger-level components until the complete software system is tested.

The second dimension of the software to be tested is the psychological functionality of the model. Does the model accurately depict human performance? The psychological functionality of the model is tested by integrating the HPP model into a system-modeling test-bed and comparing model performance to actual human operators.

Through a contract to BBN Systems and Technologies, a system-modeling test-bed called the Operability Assessment System (OASYS) is being developed. Conceptually, OASYS is a toolbox that can create "soft" prototypes of crew consoles. (A soft prototype is a physical mockup of the system in which the functional operation of the system is simulated in software.) A typical console simulated through OASYS might, in the "real world," control a satellite constellation, a nuclear power plant, or a manufacturing center. Specific tools contained in the

OASYS toolbox include requirements definition tools, rapid prototyping tools, system emulation libraries, human-in-the-loop simulation tools, data collection and analysis tools and tools to interface OMAR into OASYS in a way that allows OMAR-developed HPP models to "operate" OASYS-developed consoles.

During testing, OMAR-developed HPP models are interfaced into OASYS and employed to operate consoles. When validating an HPP model, the OASYS simulation run is designed to test the behaviors embodied in the HPP model. The results of the HPP model simulation run are compared to other OASYS simulation runs for which the console was operated by human-in-the-loop operators. This ability to perform side-by-side comparisons of operators and models is a powerful method for validating HPP models.

HPP model testing should include macro (aggregate) and micro (subcomponent) HPP model behavior. Specific behaviors to be tested and the methodology to be employed in testing depend, of course, on the psychological functionality modeled during a specific incremental development cycle. Example aggregate behaviors include task completion times and error rates. Example subcomponent behaviors include visual scanning rates and fixation pauses, short-term memory decay functions, and auditory bandwidth limitations.

Testing of an HPP model requires the availability of subjects with different skill levels to perform the tasks. Additionally, the system-modeling test-bed must be programmed with representative mission scenarios (which were identified during the requirements analysis stage). For illustrative purposes, example behaviors and testing methodologies associated with testing HPP model performance on an air defense system will be discussed. The task is very similar to a nonmilitary air traffic control system.

An air weapons controller has several unidentified aircraft approaching a restricted zone in which high value assets are being protected. The controller has two aircraft airborne and flying holding patterns. The controller's goal is to identify the unidentified aircraft. There are several options to meet this goal: (1) query the radar transponders on the approaching aircraft, (2) order the airborne aircraft to intercept the unidentified aircraft and perform a visual identification, or (3) order aircraft on the ground to launch, intercept ,and identify the incoming aircraft. These options are not mutually exclusive; the controller will probably execute at least two of them.

Variables capturing macromodel performance include items like task completion time, probability of successful identification, and method used to identify incoming aircraft. Variables capturing a visual micromodel's performance include visual scanning rate, fixation pause, and the probability of "seeing" various pieces of information. For a memory subcomponent model, variables of interest include items like recalling data in task-relevant groups (chunking) and recalling data relevant to the HPP model's interactions with the entity.

There are some very difficult methodological issues associated with the study of complex tasks. First, consider the issue of studying aggregate performance. The tasks being studied are very complex. It is difficult to get experimental subjects to respond in identical ways. Should the controller (subject) be told to execute only certain tasks in a given order? If so, data on individual differences is lost. Should the individual controllers be allowed to decide on the actions they will take and the order of those actions? If so, the data generated will vary significantly; such variance is very difficult to analyze statistically. The first case, in which the actions are controlled takes a normative approach to model validation; that is, an attempt is made to define "ideal" performance. The second case is an attempt to define a distribution. For model validation, rigorously controlling responses seems to be the correct approach. (Conversely, for system testing, allowing the operators to respond unconstrained seems to be the correct approach.)

A second difficult model validation issue is that, given that the HPP model performance diverges from the performance of actual operators, how can the HPP model subcomponent that is creating the difference be identified? Did the scanning model not see the aircraft? Did information in the short-term memory model decay too quickly? Is there a problem with the cognitive model? Thus, it is very important to validate the subcomponent models, although determining why the model does not behave like operators will still be very difficult even with validated subcomponent models.

Regardless of the problems associated with validating HPP models, the approach seems to be a major improvement over the way current models are validated. Furthermore, it goes a long way towards responding to the suggestions of the NRC studies on ways to improve the validation process for HPMs.

Operations and Maintenance

The goal of the current operability research, which includes both OASYS and OMAR test-beds, is to improve the design process for new and retrofit systems. Specifically, technology is being developed to ensure that user requirements are adequately incorporated into the design. (The slogan for this approach is "fly while you design.") The approach is to use OASYS to prototype competing system designs. These designs may differ as to the amount of automation-aiding technology, the crew-size requirements, or the skill-level requirements.

A typical study begins with the development of a "concept of operations" (CONOPS). CONOPS defines how the system will be employed, the manning levels required, and the procedures the operators will follow. The system is then tested on the representative scenarios, in accordance with CONOPS directives. Test scenarios include both normal and emergency operating to test the full range of required system performance.

Systems that meet performance requirements across the full range of scenarios are then compared on life-cycle costs. Life-cycle costs vary due to differences in the initial cost of the system and operating costs. A key driver of operating costs is manpower requirements. A system that requires more operators or operators with a greater skill level usually costs more over its life-cycle.

The system end users--the operational personnel--participate at each step. Operational personnel define system requirements and representative scenarios. Most importantly, the users "fly the system" while it is being designed, and thus provide feedback to the designers.

HPP models integrated into this process serve as team members in a multicrew environment, operating individual crew stations. The HPP models interact with the humans-in-the-loop through voice generation and recognition systems, as well as the system interface. In this capacity, HPP models reduce experimental variability by providing behavioral replication (on the part of the team members) across simulation trials and decrease study costs by reducing the number of operational personnel that need to participate in the design studies.

Although the goal is to use HPP models to analyze design options, they are currently limited to supporting roles because of their inability to accurately model human performance. Contemporary HPP models are limited in their behavioral repertoire to performing procedural tasks and, for the most part, are unvalidated. The goal is to use the method of successive approximation to develop and validate more capable models.

Validated models will eventually be archived in an object-oriented data base. The goal is to create a data base of validated models which will be available to support diverse study requirements. Some studies may require detailed visual models, other studies may not. Some

studies may require detailed cognitive models, other studies may not. The capability to selectively "compile" models of varying resolution from an existing data base of validated subcomponent models in response to specific study requirements is currently being developed.

The approach for the current research effort is to incrementally develop and validate HPP models. The effort will begin with the development of robust models of limited utility, then the models will be progressively extended through additional cycles of identifying requirements, designing enhancements to the models, developing model extensions, testing the extensions, and, finally, using the models to evaluate alternative design options for time-critical, information-dense systems.

CONCLUSIONS

Several problems associated with HPMs are discussed in the introduction. In this section, those issues are briefly reviewed and how the proposed approach to human performance modeling addresses them is briefly discussed. The first problem presented in the introduction is the lack of generalizability. Most engineering models are of limited utility; however, process models inherently seem to overcome generalizability limitations by depicting internal computational mechanisms. If the model's mechanisms are accurate, they should be applicable over a wide range of behavior.

The next issue is the lack of sophistication in depicting the structural components of the human information-processing system. Specifically, most contemporary models do not depict top-down and bottom-up processing, the parallel and serial components of the information processing system, and knowledge at multiple levels of abstraction. The holon architecture readily supports models employing multiple data streams, thus allowing the development of models with multiple processing centers simultaneously active (representing top-down, bottom-up, serial, and parallel processing). In addition, because holons are encapsulated agents, they can employ knowledge at multiple levels of abstraction.

Another issue is the lack of comprehensive models. There are a vast number of good micromodels; however, these micromodels cannot be integrated because of a lack of agreement in stages of information processing and incompatibilities between inputs and outputs of individual micromodels. Building comprehensive HPP models is an excellent way to overcome this problem because the holon architecture provides a framework for integrating micromodels. Furthermore, developing complete models initially at a low level of resolution and then incrementally adding detail is a workable approach to comprehensive model development.

Another critical issue identified by both NRC studies is insufficient model validation because of a lack of good validation techniques and effort on the part of the model developers. The approach of simultaneously building a test-bed for HPP model research (OMAR) and a system-modeling test-bed (OASYS) (with human-in-the-loop simulation capability) that can be linked together provides an important new technique for model validation and refinement. HPP models can be compared to actual humans on identical simulation runs, thus providing an easy way to validate models.

Several issues discussed in the introduction dealt with the psychological functionality represented in HPMs and the ability to "adjust" this functionality through parameters setting. In particular, the *Quantitative Modeling of Human Performance in Complex, Dynamic Systems* report (Baron et al., 1990) called for the development of models explicitly representing attention and cognitive processes. The approach of the current research effort to developing HPP models readily allows experimentation on alternative models of attention and cognitive process. The object-oriented nature of the modules easily supports substitution of one module representing

one theory with another and the comparison of both to each other and actual human performance by conducting a set of identical simulation runs with the models linked to OASYS.

Another issue for contemporary models is the lack of neurobiological plausibility. The holon architecture facilitates incorporation of cognitive neural science research results into HPMs. Individual holons are, analogously, ensembles of neurons. Each ensemble accomplishes a specific function, such as recognizing an object. Human performance, from this perspective, is the result of the cooperative action of many neuron ensembles (Edelman, 1989; Damasio, 1989). The holon architecture readily supports research on neurobiological plausible models by providing a framework to model neuron ensembles and their interactions. In particular, research can be conducted with the holon architecture to investigate the emergent properties of multiple interacting centers (representing neuron ensembles). The control logic in a holon-based model is distributed and implemented through message-passing; therefore, it is very easy to study emergent behavior.

Finally, the last issue raised in the introduction is the inaccessibility of most HPMs and the fact that most models have rarely been independently evaluated. The "Human Performance Models for Computer-Aided Engineering" (Elkind et al., 1989) report recommends that government agencies support the development of easily used versions of models on the most inexpensive machines possible. This is exactly the course of action being pursued in the current research effort. OASYS and OMAR will be made available to interested researchers through the Crew System Ergonomics Information Analysis Center (CSERIAC). CSERIAC is the information analysis center responsible for acquiring, analyzing, and disseminating technical information on crew system ergonomics. One function of CSERIAC is being the repository for computer-based models of human operators. The amount of work required to validate and extend HPP models is tremendous; therefore, an effort is being made to create and support a larger HPP model research community within government, industry, and academia.

REFERENCES

Adams, M.J., Tenney, Y.J., & Pew, R.W. (1991). *Strategic Workload and the Cognitive Management of Advanced Multi-Task Systems.* Wright-Patterson AFB, OH: Crew System Ergonomics, Information, and Analysis Center.

Allport, A. (1993). Attention and control: Have we been asking the wrong questions? A Critical review of twenty-five years. In D.E. Meyer & S. Kornblum (Eds.), *Attention and Performance XIV*. Cambridge, MA: The MIT Press.

Anderson, J.R. (1987). Methodologies for studying human knowledge. *Behavioral and Brain Sciences, 10,* 467-505.

Anderson, J.R. (1990). *The adaptive character of thought.* Hillsdale, NJ: Lawrence Erlbaum Associates.

Baron, S., Kruser, D.S., & Huey, B.M. (1990). *Quantitative modeling of human performance in complex, dynamic systems.* Washington, DC: National Academy Press.

Barsalou L. (1992). *Cognitive science: An overview for cognitive scientists.* Hillsdale, NJ: Lawrence Erlbaum Associates.

Bobrow, D.G., & Stefik, M.J. (1986). Perspectives on artificial intelligence programming. In C. Rich & R.C. Waters (Eds.), *Readings in Artificial Intelligence and Software Engineering.* San Mateo, CA: Morgan Kaufmann.

Coad, P., & Yourdon, E. (1990). Object-oriented analysis. In R.H. Thayer & M. Dorfman (Eds.), *System and Software Requirements Engineering.* Los Alamitos, CA: IEEE Computer Society Press.

Collins, A., & Smith, E.E. (1988). *Readings in cognitive science: A perspective from psychology and artificial intelligence.* San Mateo, CA: Morgan Kaufmann.

Corker, K.M. (1991). *Model development and implementation with the AIRT C2 evaluation methodology.* Unpublished working paper, Cambridge, MA.

Damasio, A.R. (1989). Time-locked multiregional retroactivation: A systems-level proposal for the neural substrates of recall and recognition. *Cognition, 33,* 25-62.

Dennett, D.C. (1978). *Brainstroms: Philosophical essays on the mind and psychology.* Cambridge, MA: The MIT Press.

Deutsch S.E., Hudlicka, E., Adams, M.J., & Feehrer, C.E. (1993). *Computational Cognitive Models: Final Report* (AL/HR-TR-1993-0027). Wright-Patterson AFB, OH: Armstrong Laboratory, Logistics Research Division.

Edelman, G.M. (1989). *The remembered present: A biological theory of consciousness.* New York: Basic Books.

Elkind J.I., Card, S.K., Hochberg, J., & Messick, B. (1989). *Human performance models for computer-aided engineering*. Washington, DC: National Academy Press.

Feehrer, C.E., & Deutsch, S.E. (1993). *Alternative Analysis for Computational Holon Architecture*. (Delivery Order #5: Technical Proposal). Cambridge, MA: BBN Systems and Technologies.

Fishwick, P. (1986). *Hierarchical Reasoning: Simulating Complex Processes Over Multiple Levels of Abstraction* (UF-CIS Technical Report TR-86-6). Gainseville, FL: University of Florida, Computer and Information Sciences Department.

Fleishman, E.A. & Quaintance, M.K. (1984). *Taxonomies of human performance: The description of human tasks*. New York: Academic Press.

Kahneman, D., & Triesman, A. (1984). Changing views of attention and automaticity. In R. Parasuraman, J. Beatty, & E. Davis (Eds.), *Varieties of Attention*. New York: John Wiley & Sons.

Kimura, D. (1992). Sex differences in the brain. *Scientific American, 267*(3),118-125 .

Koestler, A. (1967). *The ghost in the machine*. London: Cambridge University Press.

Koestler, A. (1976). *Janus: A summing up*. New York: Random House.

Kosslyn, S. M. (1984). Mental representation. In J.R. Anderson & S. M. Kosslyn (Eds.), *Tutorials in Learning and Memory*. San Francisco: Freeman.

Lachman, R., Lachman, E. C., & Butterfield, E.C. (1979). *Cognitive psychology and information processing: An introduction*. Hillsdale, NJ: Lawrence Erlbaum Associates.

Marr, D. (1982). *Vision*. San Francisco: Freeman.

Meister, D. (1985). *Behavioral analysis and measurement methods*. New York: Wiley.

Newell, A. (1980). Physical symbol systems. *Cognitive Science*,, 4, 135-183.

Shatz, C.J. (1992). The developing brain. *Scientific American, 267*(3), 60-67.

Svoboda, C.P. (1990). Structured analysis. In R.H. Thayer & M. Dorfman (Eds.), *System and Software Requirements Engineering*. Los Alamitos, CA: IEEE Computer Society Press.

Wickens, C.D. (1992). *Engineering psychology and human performance*. New York: HarperCollins.

Young, M.J. (1992). *A Cognitive Architecture for Human Performance Process Model Research* (AL-TP-1992-0054). Wright-Patterson AFB, OH: Armstrong Laboratory, Logistics Research Division.

Zeki, S. (1992). The visual image in mind and brain. *Scientific American, 267*(3), 68-77.

ACRONYMS

CAE	Computer-Aided Engineering
CLOS	Common Lisp Object System
CONOPS	Concept of Operations
CSERIAC	Crew System Ergonomics Information Analysis Center
CT	Computational Theory
HPM	Human Performance Model
HPP	Human Performance Process
MRI	Magnetic Resonance Imaging
NASA	National Aeronautics and Space Agency
NRC	National Research Council
OASYS	Operability Assessment System
OMAR	Operator Model Architecture
PET	Positron Emission Tomography
RA	Representation and Algorithm

U.S.G.P.O.:1993-550-057/81025